공학석사학위논문

Machine Learning Models and Missing Data Imputation Methods in Predicting the Progression of IgA Nephropathy

기계학습 및 결측자료 대체를 이용한 IgA 신염 예후 예측

2015 년 2 월

서울대학교 대학원

전기컴퓨터공학부 컴퓨터공학전공

노 준 혁

Machine Learning Models and Missing Data Imputation Methods in Predicting the Progression of IgA Nephropathy

기계학습 및 결측자료 대체를 이용한 IgA 신염 예후 예측

지도교수 Robert Ian McKay

이 논문을 공학석사학위논문으로 제출함

2014 년 11 월

서울대학교 대학원

전기컴퓨터공학부 컴퓨터공학전공

노준혁

노준혁의 석사학위논문을 인준함

2015 년 1 월

위 원 장 이제희 (인)

부위원장 Robert Ian McKay (인)

위 원 Bernhard Egger (인)

Abstract

Machine Learning Models and Missing Data Imputation Methods in Predicting the Progression of IgA Nephropathy

Junhyug Noh
Department of Electrical Engineering and Computer Science
College of Engineering
The Graduate School
Seoul National University

IgA Nephropathy(IgAN) occurs when IgA, an immune-system protein, deposits in kidney glomerules for unknown reasons. It is the most common glomerulonephritis, and has a high prevalence rate in East Asian nations. Determining appropriate treatment protocols and classifying IgAN patients by risk level are the most pressing issues. IgAN can occur even at a very young age (average age 35), hence the patients suffer from many personal, social and economic problems during the disease course – progression to End-Stage Renal Disease(ESRD). Although a number of approaches for predicting the prognosis of IgAN are available, well-advanced methods and techniques are scarce. In this work, we aimed to build new prediction models through careful application of machine learning methods.

Our dataset was collected from 1979 to 2014 by the Division of Nephrology, Seoul National University Hospital. It includes 1622 patients' records, with more

than 90 attributes. Among them, we chose 17 independent attributes for building our models. However, 269 records have missing values for at least one of these attributes, which can lead to a substantial loss of statistical prediction power. Hence, we used value imputation techniques to restore the records for our modelling. We used mean, mode and random imputation techniques as our baselines and analysed more sophisticated methods such as nearest neighbour hot deck imputation and Multivariate Imputation by Chained Equation(MICE). MICE with Classification And Regression Trees (CART) showed better performance, and hence we used this technique for the subsequent analysis.

With this imputed data, we explored various machine learning methods. We investigated the most popular individual learners namely CART, logistic regression and neural network, and also the ensemble learners such as bagging, random forest and boosting. We treated the problem as a classification problem, of predicting progression to ESRD within the ten years following the initial diagnosis.

All six methods yielded good classifiers, with AUC performance between 0.804 (decision tree) and 0.868 (boosting). The results were generally in-line with expectations, with poor kidney performance on presentation, and evident macroscopic and microscopic damages, all associated with poorer prognosis. Further demonstrating the benefits of the application of machine learning models in medical problems. However, a set of unexpected decision rules for a small group of patients arise some interesting questions and urge us for further detailed investigation.

Keywords: Immunoglobulin A Nephropathy (IgAN), End-Stage Renal Disease (ESRD), Missing Value Imputation, Machine Learning, Supervised Learning, Ensemble Learning

Student Number: 2013-20786

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Immunoglobulin A Nephropathy (IgAN) is the most common glomerulonephritis worldwide and the key cause of End-Stage Renal Disease (ESRD). Its clinical course is highly variable, with a 10-year renal survival rate in the range 70–80% [11]. Because patients are usually diagnosed at fairly young age, 20-30% of IgAN patients experience ESRD during their life.

Renal (kidney) function is measured by glomerular filtration rate (GFR), the volume of blood filtered from the renal glomerular capillaries per unit time. The severity of IgAN can be classified into five stages. The end of the progression is ESRD, a severe illness requiring either regular dialysis or kidney transplantation, and with poor life expectancy. The fifth stage, although it retains some kidney function, is nevertheless a severe illness, and generally progresses to ESRD. Another important stage in defining the progression is the CR2 stage, at which the serum creatinine level (a measure of the elimination effectiveness of the kidneys) has doubled.

1.1 Problem Definition

The insidious disease course and its high variability make it difficult for physicians to predict renal outcome at the time of diagnosis. This has both medical and social consequences. It is difficult for physicians to determine how aggressively to treat each individual case (as is common in medicine, aggressive treatments have more severe consequences). And it is difficult for patients to make long-term plans because of this uncertainty. Previous studies have determined some factors associated with poor renal prognosis, including initial renal function, blood pressure, and the amount of proteinuria [2]. However, they have not been able to demonstrate reliable outcome prediction. Our aim in this work is to provide more robust predictors using machine learning techniques. Specifically, we assume that we have the initial presentation and biopsy data for a patient, and aim to predict the progression to ESRD within a specific period (10 years). The outcome of IgAN progression is dichotomous (ESRD or not), and hence we have a binary prediction (classification) problem.

1.2 Motivation

By far the major challenge in the field is the identification, at an early stage, of the patients at highest risk of progression to ESRD. The tools and methods for predicting renal prognosis are limited. There is some evidence that genetic and social factors influence IgAN progression, hence it is specifically of interest to investigate progression in the relatively homogeneous Korean population.

1.3 Importance

The prevalence of glomerular diseases varies based on geographic area, race, age and other factors. Race/ethnicity is one of the risk factors for IgAN. Studies

show that IgAN is particularly prevalent and its course more severe in patients of Asian ancestry. Hence, investigation of Asian (Korean) populations can be especially effective in identifying risk factors for progression.

1.4 Contribution

Though IgAN has been widely studied in Asian countries including South Korea [19, 10], Singapore [18], China [22] and Japan [17], their research methodologies were based on traditional descriptive and exploratory statistical analysis. Hence, our proposed use of machine learning algorithms provides a useful complement, potentially useful for clinical investigations and medical and patient decision-making. This work is an extension of [24]. It is extended primarily in the following aspects:

1. we used imputation techniques to restore the missing data
2. we applied ensemble algorithms such as bagging, random forest and boosting to improve the performance
3. we analysed the results with the statistical measures such as AUC, closest topleft and Youden index

1.5 Outline of the paper

In section 2, we describe the background of IgA Nephropathy. Section 3 details our methodologies. The results are presented in section 4 and further analysed in section 5. We summarise our results in section 6.

Chapter 2

Background

2.1 Immunoglobulin A Nephropathy

Immunoglobulin A nephropathy (IgAN), first described by Berger and Hinglais [3], is the most common immune-complex-mediated glomerulonephritis (GN) – inflammation of the glomeruli of the kidney – worldwide [21, 15]. IgAN (or Berger's disease) is a chronic kidney disease in which an antibody, Immunoglobulin A (IgA), forms granular deposits in the glomeruli – blood vessels in the kidney. It is unknown why IgA is trapped in the glomuleri, but its presence causes inflammation. These mesangial IgA deposits affect the ability of the kidneys to perform their normal function of filtering waste, excess water and electrolytes from the blood.

A few IgAN patients experience complete remission, but many eventually progress to ESRD, requiring hemodialysis (for acute kidney failure) or a kidney transplant (for chronic kidney failure) for their survival. IgAN can progress slowly, over many years, through the five stages from worsening renal dysfunction to ESRD. The length of this progression varies from patient to patient, but can be from 10 to 20 years. Furthermore, even transplantation is not a com-

plete cure – in many cases, substantial mesangial IgA deposits have recurred in kidneys transplanted into patients who had developed end-stage renal disease due to IgAN [4].

2.2 Supervised Machine Learning Models

We used three widely used individual learners (Classification and Regression Trees, Logistic Regression and Neural Networks) and ensemble learners with three different techniques (Bagging, Boosting and Random Forest).

2.2.1 Individual Learners

Classification and Regression Trees (CART)

Decision tree is conceptually simple approach to classification and regression, yet is powerful. Decision trees are more expressive. It is easy to implement and interpret compare to many machine learning algorithms. It can perform better in non-linear settings. CART formulation forms a binary tree and minimizes the training error in each leaf. CART uses Gini coefficient to choose the best variable – estimates the purity of the internal nodes. Tree models represent data by a set of binary decision rules [7].

Logistic Regression

Logistic regression is based on the logistic function with a linear combination of dependent variables and is formulated as $\pi(x) = 1/1 + e^{-(\beta' X)}$ where $\beta' X = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ...$ and $\pi(x)$ as the probability $p(y = 1|X)$ – the probability that the dependent variable (y) is of class 1, given the independent variables (x_i) [12].

Neural Networks

Neural network is a bio-inspired system of programs that approximates the operation of the human brain. The inputs are represented to the neural network via the input layer. The weighted combinations of these inputs are created and put through the sigmoid function to produce the next layer of inputs – hidden layer. This next layer undergoes the same process to predict the output – output layer [26].

2.2.2 Ensemble Learners

Ensemble methods are the learning models that classifies the data by combining the results of multiple learners. They aim to improve the predictive performance of a given statistical learning model or a fitting technique. Bagging, boosting and random forest are different ensembling techniques.

Bagging

Bagging is the acronym of **b**ootstrap **agg**regat**ing**. It builds the predictors by using repeated bootstrap samples from training dataset, and aggregates those predictors. For aggregation, the average is used for regression model and plurality vote for classification model. We chose CART as a base learner among various algorithms [5].

Random Forest

Random forest algorithm adds more randomness to bagging. It uses a decision tree as base learner, however the way of splitting is different from the standard decision tree. At each node, it chooses a specific number of attributes randomly and finds the best split among them [6].

Boosting

The first developed boosting algorithm is AdaBoost and its base predictors have different weights for each observation. The weight of misclassified observation is increased and weight of opposite is decreased in each step, and the final predictor is obtained by aggregating all base predictors [27]. AdaBoost can be interpreted as a gradient descent algorithm, which updates the base learner to decrease loss function. There are various boosting algorithms available in literature with different loss functions and base learners. In our research, we used the negative binomial log-likelihood as the loss function and the generalized additive model as the base learner [8].

Chapter 3

Methods

3.1 Dataset

The dataset has been built up by the Division of Nephrology, Seoul National University Hospital (SNUH) – one of the best-reputed hospitals in South Korea. It details 1622 Korean biopsy-confirmed IgAN patients who were identified between the years 1979 and 2014. The dataset was last updated on May 29th, 2014. Most patients' biopsy tests were analysed by the same laboratory; in the exceptional cases, appropriate corrections were made to retain consistency.

The dataset consists of data about the patients' initial presentation and biopsy, and their GFR information from subsequent follow-up sessions. The dataset includes 91 attributes, grouped into four categories : demographic, laboratory, clinical and histological. The input attributes in our binary classification predictive modelling, come from the initial presentation data; the GFR values measured during the follow-up sessions are not used for the modelling. However, the value of the target attribute, ESRD, also depends on the follow-up GFR data, and we plan to investigate its use for updated predictions in subsequent work.

Table 3.1: Attributes Used for Modelling

Category	**Type**	**Name**	**Description**
Demographic	Continuous	AGE	age of patient
	Dichotomous	SEX	sex of patient
Histologic	Continuous	GLOM	no. of glomeruli
		CRES%	% of crescent
		GS%	% of global glomerulosclerosis
		SS%	% of segmental glomerulosclerosis
	Ordinal	IF	renal tubule fibrosis
		TA	renal tubule atrophy
		II	renal tubule infiltrate (inflammatory)
Clinical	Continuous	SBP	systolic blood pressure
		BMI	body mass index
	Ordinal	SMHX	smoking history
Laboratory	Continuous	HB	hemoglobin
		ALB	serum albumin
		CHOL	cholesterol
		GFR	glomerular filtration rate
		PU	24 hours proteinuria

3.2 Attributes Used for Modelling

Among the 91 attributes, we relied on the domain knowledge of the nephrologists to choose 17 independent attributes (refer Table 3.1) to build machine learning models. They are AGE, SEX, GLOM, CRES%, GS%, SS%, IF, TA,

II, SBP, BMI, SMHX, HB, ALB, CHOL, GFR and PU.

GFR is computed with the standard Modification of Diet in Renal Disease (MDRD) equation [20], adapted for Koreans:

$$\begin{aligned} \text{GFR} = 175 \quad &\times \quad \text{AGE}^{-0.203} \times \text{CR}^{-1.154} \\ &\times \quad 1.0 \text{ (if male)} \\ &\times \quad 0.742 \text{ (if female)} \end{aligned} \tag{3.1}$$

where GFR is measured in ml/min/1.73m^2. The normal GFR value is above 90ml/min/1.73m^2 with no proteinuria. If the GFR is very low ($<$ 15ml/min/1.73m^2), the patient is more likely to progress to ESRD. In the equation 3.1, CR is the creatine level.

3.3 Missing Value Imputation

Initially, the medical records were maintained manually, and there are missing values in those older records. Thus, the missing values mainly depend on the patient's first-visit date – the records were computerised in 1999, after which missing values are rare. Among 17 independent variables, there are average 0.05 missing values for patients from 1999 and 2.12 missing values for patients before 1999.

However, the first-visit date is not used for modelling. Thus, the nature of the missing data relative to our learning task is MCAR (Missing Completely At Random) and we can use complete case analysis without incurring bias.

With complete case analysis, we are limited to discard 269 records which have missing values for at least one attribute. This leads to a substantial loss of statistical power. To overcome this issue, we used imputation techniques to restore the records for our modelling.

Table 3.2 shows distribution of missing values by attributes and patients. The attribute SEX (one of the modelling attributes) does not have any missing

Table 3.2: The Number of Missing Values by Patient

Missing	Frequency	Percentage	Cumulative Frequency	Cumulative Percentage
0	1353	83.42	1353	83.42
1	108	6.66	1461	90.07
2	19	1.17	1480	91.25
3	14	0.86	1494	92.11
4	7	0.43	1501	92.54
5	7	0.43	1508	92.97
6	10	0.62	1518	93.59
7	14	0.86	1532	94.45
8	9	0.55	1541	95.01
9	3	0.18	1544	95.19
10	42	2.59	1586	97.78
11	21	1.29	1607	99.08
12	6	0.37	1613	99.45
13	3	0.18	1616	99.63
14	0	0.00	1616	99.63
15	5	0.31	1621	99.94
16	1	0.06	1622	100.00

values in the whole dataset and hence it was excluded from the imputation process. It is harder to impute the records with many missing values close to the real values, which can also introduce huge bias in the model. This urges us to allow the maximum number of attributes which can have missing values for imputation process. We fix 7 as our reasonable choice for the number of attributes which can have missing values. By this, we can restore 179 cases and add to complete cases.

For restoring these 179 cases, we evaluated various imputation methods using R's libraries: HOTDECKIMPUTATION [1, 14] (hot deck imputation) – Nearest Neighbour and MICE [9] (Multivariate Imputation by Chained Equation) – Predictive Mean Matching (PMM), CART, etc.. We chose the best method (MICE - CART with supplemented attributes) and imputed the missing values for 16 independent modelling attributes.

We introduced missing values randomly in the test set of 1363 complete cases. Original dataset has 83.4% complete cases, hence we did not introduce missing values in the test set proportionally. We introduced from 10% to 50% missing values randomly in each attribute after preserving 20% and 50% of complete cases. Substantially, we were left with 10 testsets. The two criteria (refer 3.2 and 3.3) were used to evaluate the performance of imputation methods.

To measure the performance of imputation methods, two criteria

$$\text{Criteria 1 (Normalized L1 Distance)} = \frac{1}{N}\sum_{i=1}^{n}\sum_{j\in\mathbf{M}_i}|\frac{x_{ij}-\hat{x}_{ij}}{sd_i}| \quad (3.2)$$

$$\text{Criteria 2 (Normalized L2 Distance)} = \frac{1}{N}\sum_{i=1}^{n}\sum_{j\in\mathbf{M}_i}(\frac{x_{ij}-\hat{x}_{ij}}{sd_i})^2 \quad (3.3)$$

were used where i denotes the index of attribute and $\mathbf{M}_i$ denote the index set of observations which include at least one missing value.

We also compared the distributions (mean, standard deviation) of attributes for the original and imputed data. This process allowed 1532 records from 1622 records(94.5%).

3.4 Target Attribute

The target attribute, ESRD, is a binary variable taking values 0 (negative class indicating the absence of ESRD – non-ESRD) and 1 (positive, the presence of

ESRD). From the original dataset, we remove any records missing ESRD status (labels), leaving 1528 medical records.

3.5 Setting Prediction Period

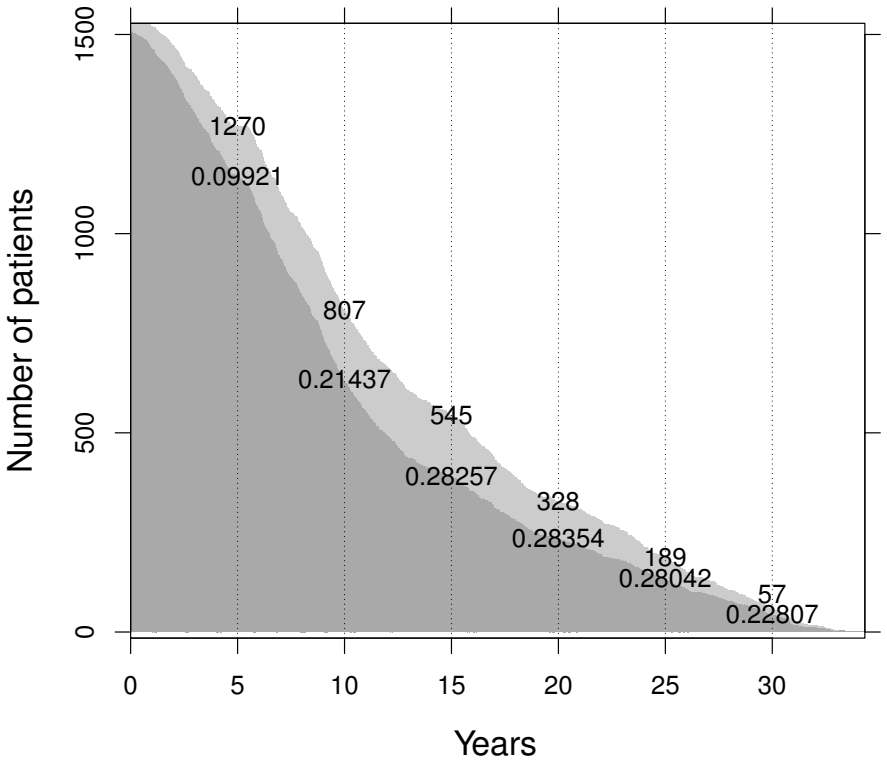


Figure 3.1: Total Cases and Positive Ratio vs Years since Biopsy
(Dark Grey: Negative Cases; Light Grey: Positive Cases)

In medical practice, 5- and 10-year survival rates are generally used for estimating the prognosis of a disease. 5-year survival is more useful in aggressive diseases with a shorter life expectancy following diagnosis, whereas 10-year is more practical in less invasive diseases with a long life expectancy. Following this

model, ESRD progression is also generally expressed by 5- and 10-year renal survival. Using standardised periods is important for understanding disease severity and comparing treatment effectiveness.

Exploratory data analysis was used to identify the most suitable target period for prediction. If we chose too short a period, positive cases would be too rare, and predictions would be of limited value. On the other hand, because the data is still being accumulated, a long period would have too few overall cases. Figure 3.1 shows this graphically. Against the prediction period, we plotted the number of patients whose records cover that period (a patient's records are considered to cover a period of N years if the interval between the patient's initial biopsy date and the last database update is at least N years). We divided the patients into those who had not reached ESRD after N years (dark grey) and those who had (light grey). The whole numbers on the plot are the total number of patients in the sample, while the real numbers are the proportion of positive cases. From the figure, we concluded that 5 years was too short to be useful (too few positive cases), while 15 was too long (too few overall cases). Thus, data properties confirmed our choice of 10 years.[1] This left 807 cases for modelling.

After setting the period, we need to consider some patients who were dead before reaching ESRD. If the difference between the first biopsy date and the DEATH date is greater than 10 years, we can use that record. However, the difference is less than 10 years, we don't know the exact ESRD value after 10 years from biopsy date. After removing these records, we finally arrived at a cohort of 785 patients' data for our modelling.

[1] We assume that the ESRD value is 0 until the ESRD date (the date on which ESRD is confirmed as 1). Specifically, this means that if the difference between the first biopsy date and the ESRD date is greater than 10 years, we consider those cases as non-ESRD (ESRD = 0).

3.6 Data Partitions

The 785 records divide into 612 negative (ESRD = 0) and 173 positive (ESRD = 1). We split the 785 records into two disjoint datasets: a training dataset (600 records) and a test dataset (185 records) by stratified random sampling. We also formed a third "mixed" dataset of records covering less than 10 years. The mixed dataset includes 711 patients: 658 cases without ESRD and 63 with. ESRD is irreversible, so cases with ESRD (= 1) before 10 years also be positive after 10 years. For the 63 positive cases in the mixed dataset, we can validate the prediction of the models (true positives). But cases without ESRD (= 0) now may progress to ESRD (= 1) within 10 years. The 658 negative cases can be predicted by the model (illustrating usefulness), but cannot validate it.

We built the binary classifiers using the training dataset. We applied the prediction models to the held-out test dataset to analyse the performance of the classifiers. Finally, we both validated (for ESRD = 1 cases) and predicted (for ESRD = 0 cases) the ESRD stage after 10 years for the observations in the mixed dataset, using the classifiers.

3.7 Building Models and Parameter Tuning

We built classifiers using by individual and ensemble learning algorithms using R's statistical modelling tools and libraries [29]. We used RPART [28] (classification tree), GLM [12] (logistic regression), NNET [26] (neural network) for individual learning and IPRED [25](bagging), RANDOMFOREST [23](random forest), MBOOST [8](boosting) for ensemble learning. We used cross validation to avoid overfitting and tune model parameters. The parameters and values used for learning are described in section 4.

For each method, we first defined the sets of learning parameter values. For each set of parameters, we performed 5-fold cross validation, splitting the training set (600 records) into 5 cross validation folds [16] by random sampling

(the same 5 folds were used throughout). One among the 5 folds was held out, and the model was fitted to the remainder. It was used to predict the held-out fold. This was repeated for each fold. We then computed the average prediction performance across folds.

We used Receiver Operating Characteristic (ROC) analysis to choose the best parameter set, computing the average Area Under the ROC Curve (AUC) across the 5 cross validation sets. AUC is an effective measure to find the best candidate model: larger AUC is better. Then we re-trained using these parameters on the full training set.

3.8 Performance Evaluation of Models

We assessed the performance of the trained models by applying them to the held-out test dataset to predict ESRD. We again used the ROC curve analysis to evaluate the performance (discriminatory ability) of the different learning models. ROC is a plot of Sensitivity (true positive rate) against (1− Specificity) (false positive rate). We used the ROC plot to detect two best cut-off points. One is the Closest TopLeft cut-off (CTL) which is on the ROC curve closest to the coordinate (0, 1) – i.e. nearest to the upper left corner of the ROC plot. The other is Youden Index cut-off (YI) which maximise total accuracy. We compared the performance of the models by estimating AUC.

$$\text{CTL} = \underset{\text{cut-off}}{\text{argmin}} \sqrt{(1 - Sensitivity)^2 + (1 - Specificity)^2} \tag{3.4}$$

$$\begin{aligned} \text{YI} &= \underset{\text{cut-off}}{\text{argmax}}\ n_{pos} Sensitivity + n_{neg} Specificity \\ &= \underset{\text{cut-off}}{\text{argmax}}\ Accuracy \end{aligned} \tag{3.5}$$

3.9 Validation and Prediction of Models

63 cases in the mixed dataset have known disease status (ESRD = 1). They can assess the model's ability to correctly identify positives cases (ESRD = 1) using the cut-off point which is the closest to the top-left of the plot. Our priority of analysis, on the mixed dataset, is on sensitivity.

Chapter 4

Results

4.1 Missing Value Imputation

In this section, we discuss the performance evaluation of the imputation methods on the test set. The imputation performance is measured by the two criteria as defined in the equations 3.2 and 3.3.

4.1.1 Mean, Mode, Random Imputation

Mean, mode, and random imputation are the simplest methods and we used these as our baseline methods. In case of mean imputation, we did not use mean value itself as imputed value but we used the observed value which was the closest to mean. Table 4.2 shows the performance of the three imputation methods on the test set.

Mean and mode imputation do not have large bias relatively. But they can distort distribution because the variance tend to be underestimated by imputing every missing values to one value.

Table 4.1: Performance of Mean, Mode, Random Imputation

Criteria	Method	Missing Ratio				
		0.1	0.2	0.3	0.4	0.5
1	Mean	0.7509	0.7391	0.7163	0.7031	0.6999
	Mode	0.8109	0.8264	0.7981	0.7881	0.7699
	Random	1.0002	0.9736	0.9679	0.9600	0.9353
2	Mean	2.1839	1.6169	1.3615	1.1211	1.1402
	Mode	2.6920	2.1657	1.8750	1.7259	1.6115
	Random	3.0554	2.4969	2.3436	2.2190	2.0644

(a) Preserving 50% of Complete Cases

Criteria	Method	Missing Ratio				
		0.1	0.2	0.3	0.4	0.5
1	Mean	0.7084	0.7036	0.7017	0.6954	0.6944
	Mode	0.7630	0.7907	0.8016	0.7976	0.7923
	Random	0.9203	0.9413	0.9503	0.9271	0.9422
2	Mean	1.0207	0.9708	0.9772	0.9710	1.1013
	Mode	1.4538	1.5197	1.5551	1.5412	1.6534
	Random	1.8020	1.9477	1.9172	1.8483	2.0096

(b) Preserving 20% of Complete Cases

4.1.2 Nearest Neighbour (NN) Hot Deck Imputation

Nearest neighbour hot deck imputation method finds the most similar record to the record which has missing values. We used Manhattan and Euclidean distance to measure the level of closeness. However, every attribute has different distribution and hence we normalised values by standard deviation or range.

Imputation using Manhattan distance shows the best performance, but it is not comparable to the baseline imputation methods owing to weight. We used same weights to all attributes, but the relative importance and relation to other attributes can be different for any specific attribute. Hence, we need to adjust weights based on domain knowledge or data analysis.

Table 4.2: Performance of NN Hot Deck Imputation

Criteria	Method		Missing Ratio				
			0.1	0.2	0.3	0.4	0.5
1	Eucl	Range	0.7858	0.7423	0.7324	0.7529	0.7508
		SD	0.8017	0.7703	0.7633	0.7590	0.7521
	Man	Range	0.7180	0.7116	0.7103	0.6961	0.7235
		SD	0.7406	0.7124	0.6915	0.7088	0.7211
2	Eucl	Range	2.4971	1.8076	1.5808	1.6046	1.5404
		SD	2.5649	1.8878	1.6649	1.5823	1.5140
	Man	Range	2.3645	1.7972	1.5853	1.4770	1.4874
		SD	2.5185	1.8049	1.5544	1.6015	1.5268

(a) Preserving 50% of Complete Cases

Criteria	Method		Missing Ratio				
			0.1	0.2	0.3	0.4	0.5
1	Eucl	Range	0.7438	0.7815	0.7800	0.7833	0.7841
		SD	0.7462	0.7768	0.7832	0.7970	0.7921
	Man	Range	0.6965	0.6903	0.7169	0.7299	0.7479
		SD	0.6804	0.6943	0.7243	0.7241	0.7512
2	Eucl	Range	1.3779	1.4639	1.4487	1.4685	1.5948
		SD	1.3484	1.4212	1.4514	1.5182	1.6317
	Man	Range	1.3033	1.2174	1.3431	1.3515	1.5478
		SD	1.2103	1.2297	1.3442	1.3314	1.5308

(b) Preserving 20% of Complete Cases

4.1.3 Multivariate Imputation by Chained Equation (MICE)

MICE imputes repeatedly using Gibbs sampling. It estimates formula for each attribute which is computed by other attributes and each attribute can be handled by different method for it. Hence, it can solve the issue arising from NN Hot Deck imputation method. Table 4.3 shows the three methods which we used.

As mean imputation, it finds the closest value from estimated value and use that as the imputed value. Table 4.4 shows the performance of MICE.

Imputation using CART shows the best performance. However, it is not better than baseline methods. The reason is standard of choosing independent attributes for modelling. Our 17 chosen modelling attributes are relatively

Table 4.3: Methods for MICE

	Type	**Method**
PMM	All	Predictive mean matching
Mixed	Numeric	Predictive mean matching
	Ordinal	Proportional odds model
	Dichotomous	Logistic regression
	Nominal	Polytomous logistic regression
CART	All	Classification and regression trees

Table 4.4: Performance of MICE

Criteria	**Method**	**Missing Ratio**				
		0.1	0.2	0.3	0.4	0.5
1	PMM	0.7361	0.7224	0.7393	0.7394	0.7582
	Mixed	0.7208	0.7174	0.7108	0.7560	0.7809
	CART	0.6901	0.6992	0.6896	0.7123	0.7234
2	PMM	2.4575	1.8977	1.7930	1.5504	1.5731
	Mixed	2.2937	1.8905	1.5707	1.6593	1.6582
	CART	2.3023	1.8002	1.5743	1.5248	1.5094

(a) Preserving 50% of Complete Cases

Criteria	**Method**	**Missing Ratio**				
		0.1	0.2	0.3	0.4	0.5
1	PMM	0.6979	0.7667	0.7365	0.7558	0.7860
	Mixed	0.6920	0.7431	0.7414	0.7622	0.7774
	CART	0.6744	0.7106	0.7016	0.7065	0.7441
2	PMM	1.2603	2.7880	1.5219	1.4187	1.5913
	Mixed	1.1709	1.7048	1.2800	1.4009	1.5928
	CART	1.2764	1.9967	1.2903	1.4304	1.5086

(b) Preserving 20% of Complete Cases

independent to each other. There is no correlation among attributes – no multicollinearity – and hence our approach did not perform better.

4.1.4 MICE with Supplemented Attributes

To solve the above problem, we also added other attributes in the original dataset, which are not used for modelling. Except the attributes those have data types as text and have many missing values, all the remaining 46 attributes

were added for imputation.

Table 4.5: Performance of MICE with Supplemented Attributes

Criteria	Method	Missing Ratio				
		0.1	0.2	0.3	0.4	0.5
1	PMM	0.5958	0.6546	0.6193	0.6321	0.6446
	Mixed	0.7177	0.6417	0.6399	0.6512	0.6732
	CART	0.6514	0.5938	0.6068	0.6252	0.6321
2	PMM	1.9663	1.6236	1.2989	1.2393	1.2500
	Mixed	2.2948	1.6019	1.3520	1.2837	1.3574
	CART	2.1455	1.4753	1.3141	1.2742	1.2623

(a) Preserving 50% of Complete Cases

Criteria	Method	Missing Ratio				
		0.1	0.2	0.3	0.4	0.5
1	PMM	0.6178	0.6187	0.6398	0.6466	0.6736
	Mixed	0.6331	0.6342	0.6603	0.6626	0.6800
	CART	0.5560	0.5827	0.6007	0.6188	0.6237
2	PMM	1.0267	1.0103	1.0593	1.0411	1.2574
	Mixed	1.1025	1.0095	1.1011	1.0985	1.2818
	CART	0.8968	0.9306	0.9828	1.0262	1.1631

(b) Preserving 20% of Complete Cases

When we used CART method, the performance was enhanced in most cases. But for cases with have many missing values, the second criteria had worse values although first criteria was good. This implies that some imputed values were very far from the real value and hence the square of distance became very large. However, we set maximum number of missing values per record as 7 to avoid this larger bias.

From the analyses of the above imputation results, we chose MICE using CART as imputation method for our data and restored 179 records.

4.2 Modelling with Individual Learning

We discuss the values chosen for model parameters, training results, performance evaluation of the classifiers on the test set, and validation and predic-

tion on the mixed dataset. The classifier performance is illustrated visually with ROC plots and analysed with statistical measures: sensitivity, specificity, accuracy and AUC.

4.2.1 Classification & Regression Trees (CART)

We created the classification trees using the RPART [28, 29] (Recursive PARTitioning) routines of R, which implement CART [7]. RPART explores the attributes and threshold values for splitting, choosing the decision rules which minimise classification impurity using the Gini index.

Choosing the CART Parameters:

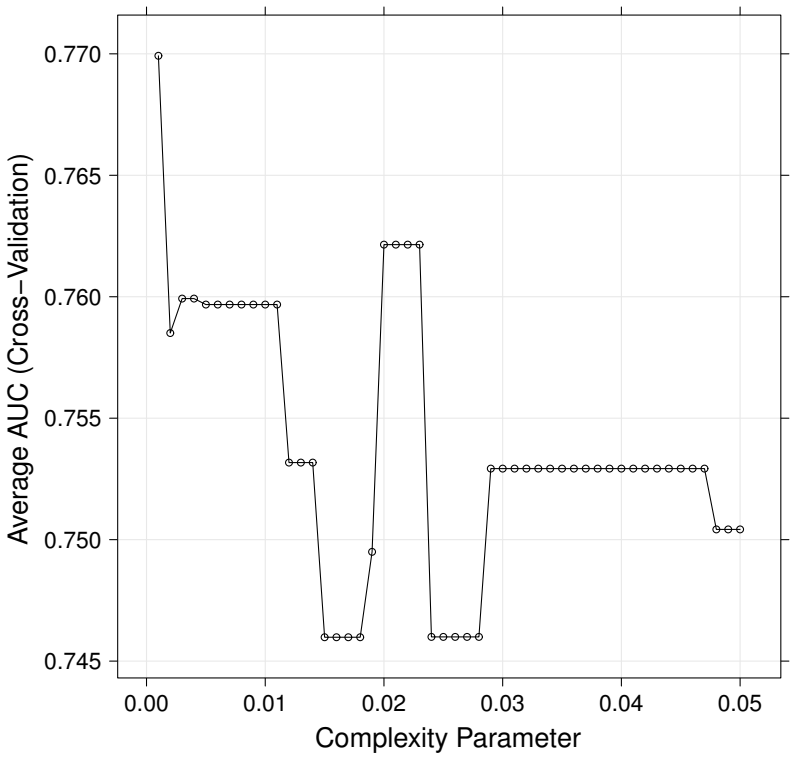


Figure 4.1: Average AUC Vs. Complexity Parameter

The main parameter of RPART is complexity parameter (**cp** – controls

pruning of splits). We chose model parameters using cross-validation as described in sub-subsection 3.7.

The average AUC across the 5-fold cross validation sets for complexity parameters (cp) in the range $[0.00, 0.05]$ is plotted in Figure 4.1. The settings $cp = 0.001$ gave the highest average AUC ($= 0.770$), so they were used in the rest of the analysis.

Training Dataset Results:

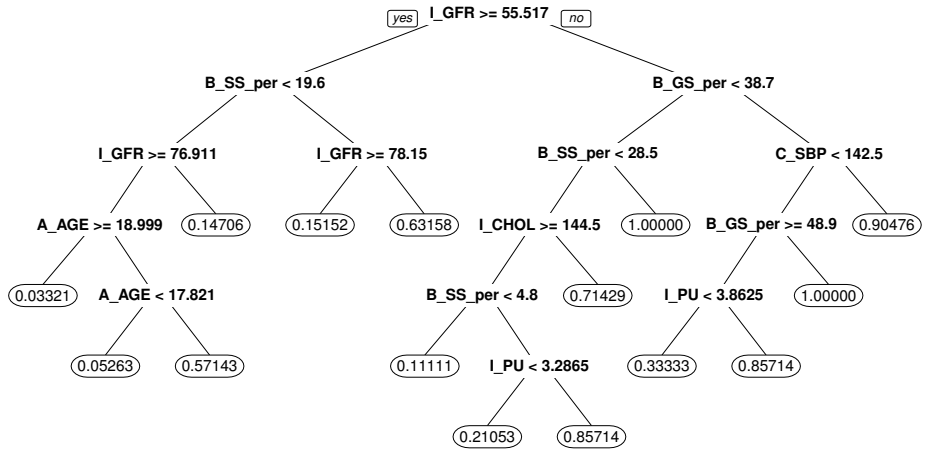


Figure 4.2: Classification Tree from Chosen Parameter Settings: Complexity Parameter= 0.001

Figure 4.2 shows the binary classification tree derived from the training set ($cp = 0.001$). Internal node labels are attributes, edge labels are attribute threshold values for the split, and leaf (terminal) nodes show the sample relative frequency of ESRD = 1, given the decision rule. The AUC for this tree was 0.770.

Evaluation on Test Dataset:

In Figure 4.3, the ROC curve rises well above the diagonal, indicating good model performance. The largest AUC (shaded in grey) had area 0.804. The closest topleft cut-off for predicting ESRD stage was determined as 0.1291. Probabilities (leaf node values) below 0.1291 are classified negative (ESRD = 0), and the rest positive (ESRD = 1). At this cut-off, with distance = 0.3812, sensitivity was 0.8780 and specificity 0.6389 (1− specificity = 0.3611).

The Youden index cut-off was 0.1810. At this cut-off, with accuracy = 0.8000, sensitivity = 0.6098 and specificity = 0.8542 (1− specificity = 0.1458).

Validation and Prediction on Mixed Dataset:

We used the model to predict ESRD values for the mixed dataset. The decision tree validated 55 cases as true positive (ESRD = 1) with sensitivity = 0.8730. It predicted 311 among the 658 unknown cases to progress to ESRD = 1 within 10 years.

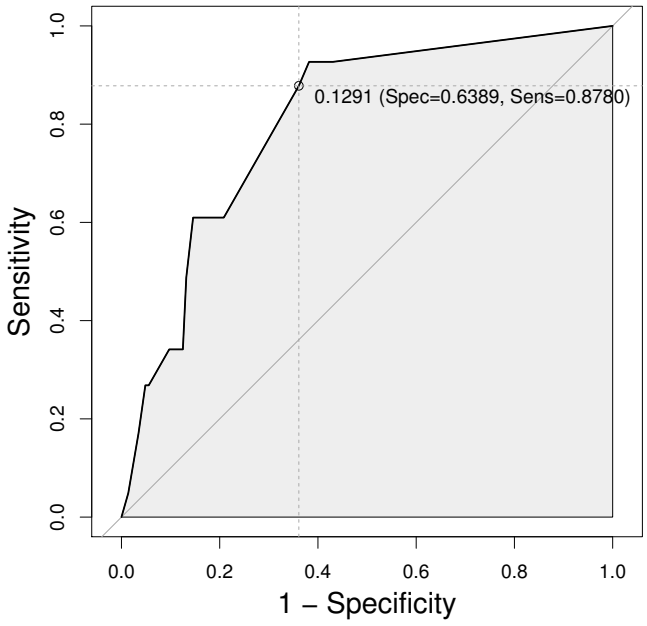
4.2.2 Logistic Regression

We used the GLM [12, 29] library of R to fit the logistic regression model. It is a generalised linear model (GLM) using a binomial distribution for the response with a logit link function. We used p-values < 0.05 to identify significant attributes.

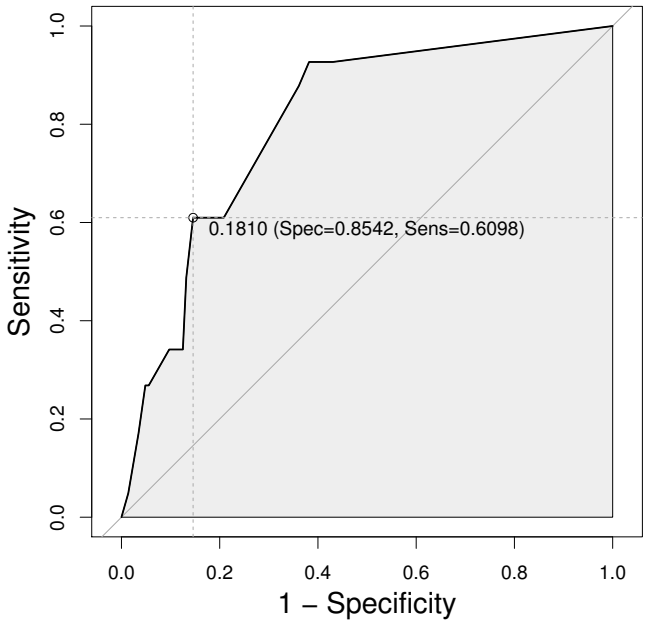
Stepwise Variable Selection:

We used stepwise variable selection to choose the most important variables. This adds or removes variables repeatedly, improving the model at each step. If there is no available improvement by adding or subtracting variables, the algorithm stops and returns the new model.

We used AIC (Akaike's information criterion) [29], often used as the model



(a) Closest Topleft Cut-off Point



(b) Youden Index Cut-off Point

Figure 4.3: ROC plot for Decision Tree (CART)

Table 4.6: Logistic Regression with Variable Selection

Name	Coefficient	p-value
Intercept	1.0154	0.3024
GFR	−0.0251	$1.40e-07$
SS%	0.0543	$3.23e-06$
GS%	0.0262	$4.49e-05$
HB	−0.2408	0.0003
SMHX	0.4297	0.0130
IF	1.3291	0.0220
SEX	0.6695	0.0307
II	−0.3462	0.0371

selection criteria for GLM, to fit the model. The procedure for deletion or inclusion is based on AIC, defined as (-2 maximised log-likelihood + 2 number of attributes). It stops when the AIC cannot be improved.

The model selected the significant attributes (p-values < 0.05) as GFR, SS%, GS%, HB, SMHX, IF, SEX, and II (refer Table 4.6). With this variable selection, the AUC was 0.852.

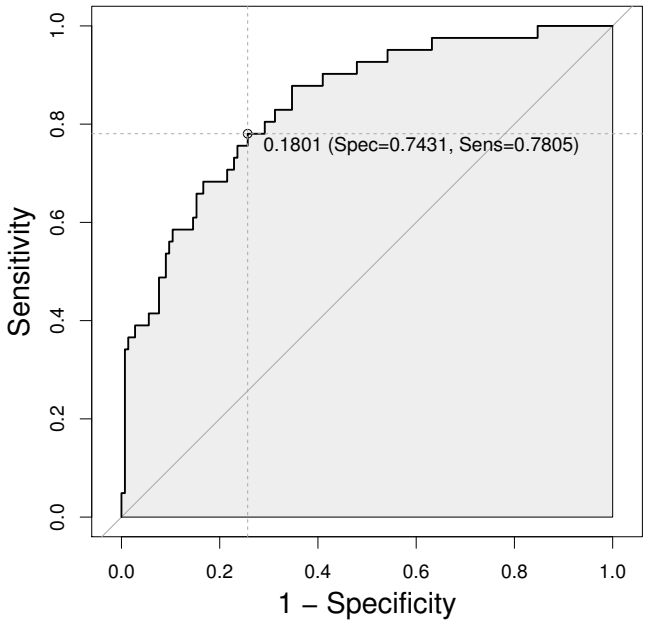
Evaluation on Test Dataset:

For logistic regression, the largest AUC was 0.840(Figure 4.4). The closest topleft cut-off was 0.1801, sensitivity was 0.7805 and specificity 0.7431. The distance from the topleft part of the plot was 0.3379.

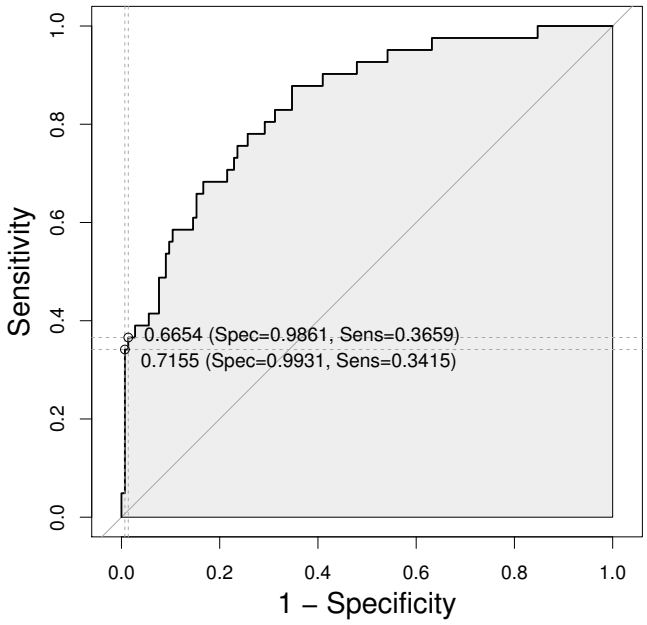
There were two Youden index cut-off points and they were 0.6654 and 0.7155. At these cut-offs, accuracy was 0.8486. By the first cut-off, sensitivity was 0.3659 and specificity 0.9861 (1− specificity = 0.0139). By the second cut-off, sensitivity was 0.3415 and specificity 0.9931 (1− specificity = 0.0069).

Validation and Prediction on Mixed Dataset:

Logistic regression correctly identified 62 instances from the mixed dataset as positive cases, with sensitivity = 0.9841. It predicted that 285 cases would progress to ESRD = 1 within 10 years.



(a) Closest Topleft Cut-off Point



(b) Youden Index Cut-off Point

Figure 4.4: ROC plot for Logistic Regression

4.2.3 Neural Networks

We used the NNET [26, 29] package in R, which builds feed-forward neural networks with a single hidden layer.

Choosing the Model Parameters:

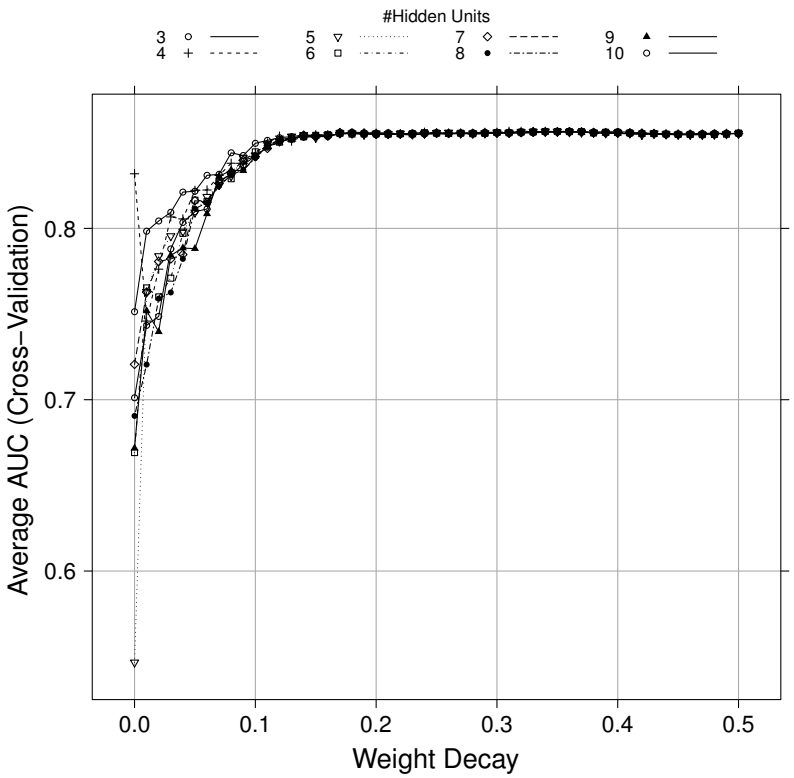


Figure 4.5: Average AUC Vs. Decay and Size

The **decay** parameter ensures that the model does not overtrain, and the **size** parameter specifies the number of nodes in the hidden layer. From Figure 4.5, we observe that the best model (AUC = 0.857) has 4 hidden layer nodes and a decay parameter of 0.34.

Table 4.7: Neural Network: Attribute Importance

Name	Attribute Importance
GFR	19.4231
SS%	14.0857
GS%	11.8157
HB	11.6184
IF	8.5895
SMHX	4.8567
GLOM	4.1586
ALB	4.0644
II	3.5029
SBP	3.0601
BMI	2.7769
SEX	2.7458
PU	2.5762
TA	2.3870
CHOL	2.3220
CRES%	1.6447
AGE	0.3724

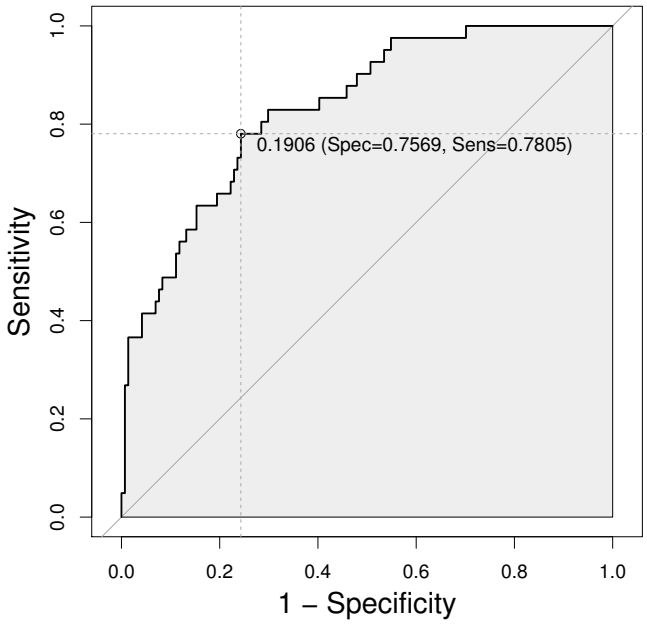
Training Dataset Results:

We used all 17 attributes in the input layer, and a hidden layer of 4 nodes. The relative importance of the 17 input variables are listed in Table 4.7. The relative importance was computed with Garson's algorithm [13], which determines the overall influence of each predictor variable. The most important attributes were GFR, SS%, GS%, and HB.

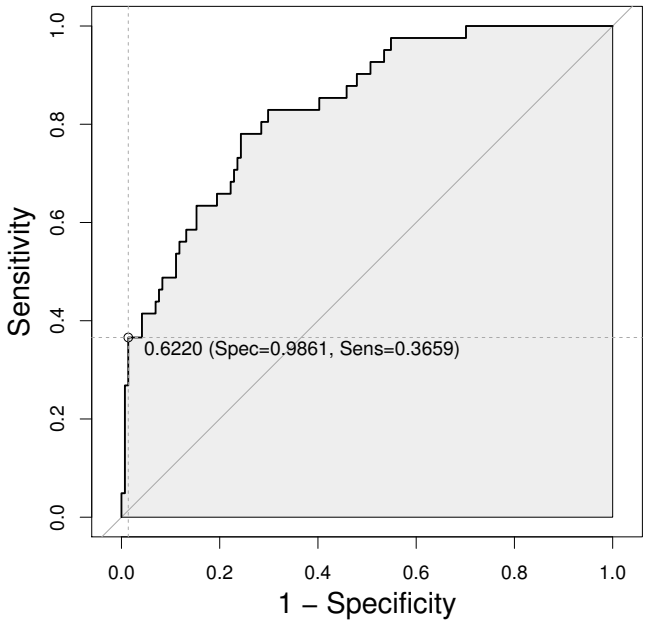
Evaluation on Test Dataset:

For neural network, the AUC was 0.834(Figure 4.6). We determined the closest topleft cut-off as 0.1906. At this cut-off, the sensitivity was 0.7805 and specificity 0.7569 (1− specificity = 0.2431), the distance was 0.3275.

The Youden index cut-off was 0.6220. At this cut-off, accuracy was 0.8486, with sensitivity = 0.3659 and specificity = 0.9861 (1− specificity = 0.0139)



(a) Closest Topleft Cut-off Point



(b) Youden Index Cut-off Point

Figure 4.6: ROC plot for Neural Network

Validation and Prediction on Mixed Dataset:

The neural network predicted 217 cases to progress to ESRD = 1 within 10 years. It validated 62 cases as true positives, with sensitivity = 0.9841.

4.3 Modelling with Ensemble Learning

Ensemble models make predictions by combining the results of multiple individual learners. Theoretically, ensemble learning methods improve the performance of the predictor. In the following sub sections, we discuss the results of three ensemble learning techniques –bagging, boosting and random forest.

4.3.1 Bagging

We used the IPRED [25, 29] package in R, which builds bagged CART.

Training Dataset Results:

Our bagging model combined 25 classification trees which were built on different bootstrap samples. We observed average AUC for training set was 0.8166.

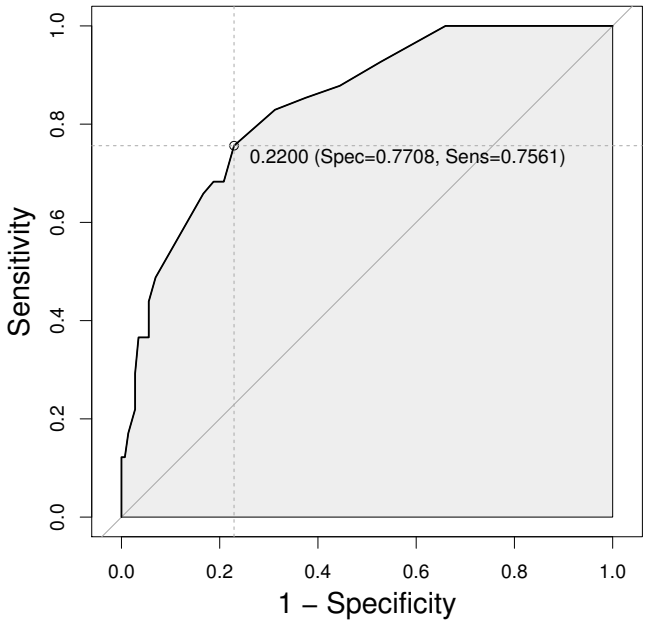
Evaluation on Test Dataset:

For bagged CART, the AUC was 0.841(Figure 4.7). We detected the closest topleft cut-off as 0.2200. At this cut-off, the sensitivity was 0.7561 and specificity 0.7708 (1− specificity = 0.2292), the distance was 0.3347.

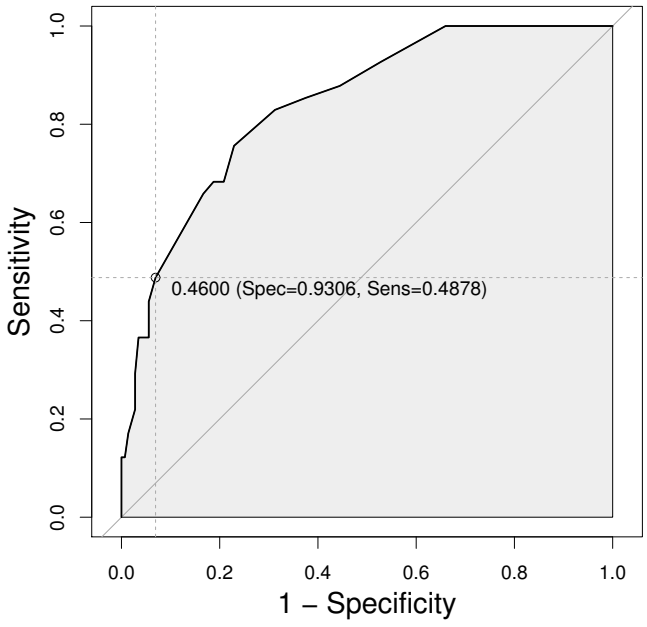
The Youden index cut-off was 0.4600. At this cut-off, accuracy was 0.8324, with sensitivity = 0.4878 and specificity = 0.9306 (1− specificity = 0.0694)

Validation and Prediction on Mixed Dataset:

The bagged CART predicted 212 cases to progress to ESRD = 1 within 10 years. It validated 59 cases as true positives, with sensitivity = 0.9365.



(a) Closest Topleft Cut-off Point



(b) Youden Index Cut-off Point

Figure 4.7: ROC plot for Bagging

4.3.2 Random Forest

We used the RANDOMFOREST [23, 29] package in R, which builds random forest model.

Choosing the Model Parameters:

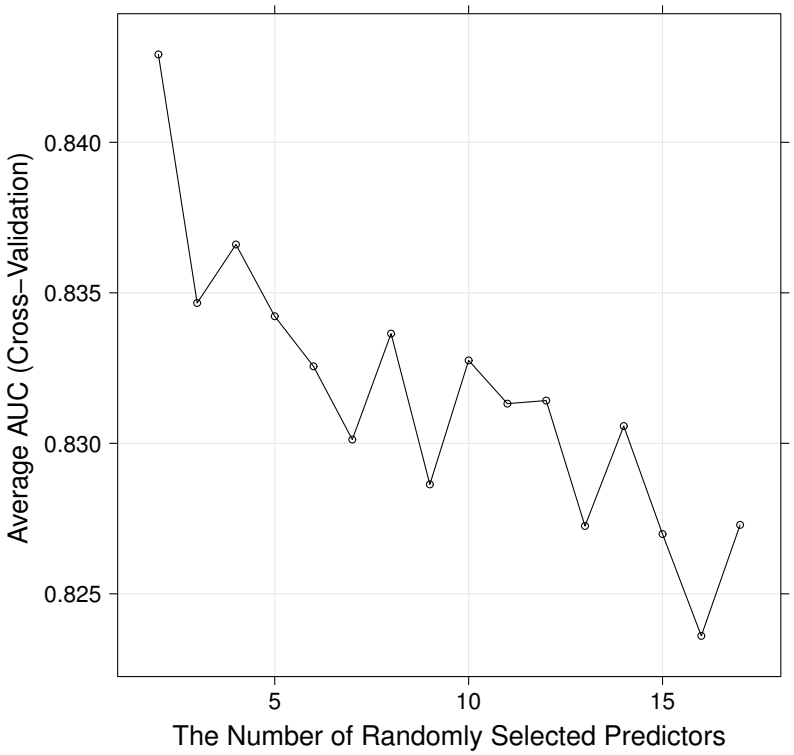
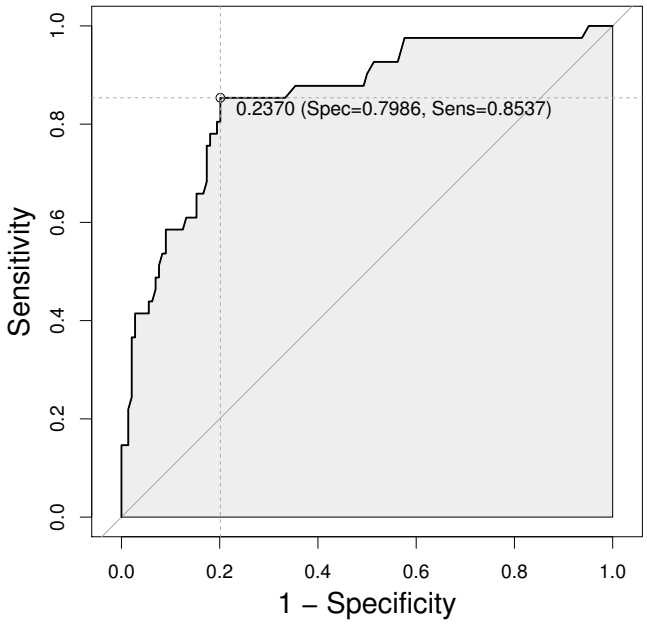
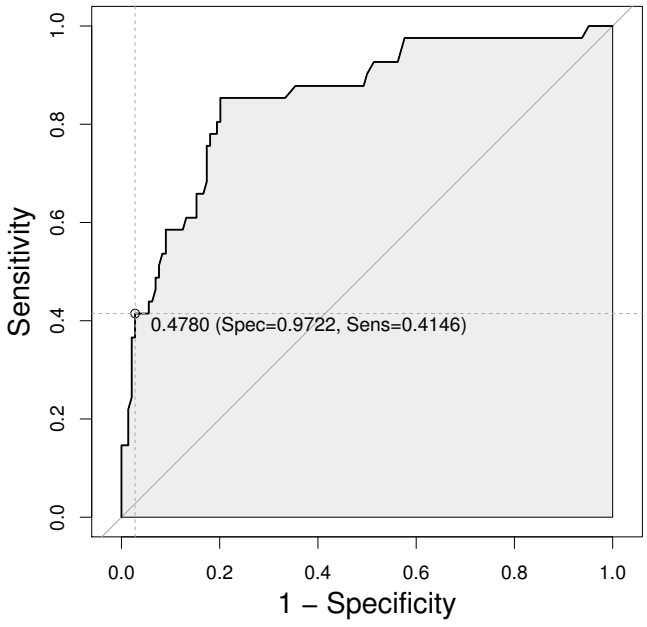


Figure 4.8: Average AUC Vs. Number of Randomly Selected Predictors

Our random forest model combined 500 classification trees. The **mtry** parameter specifies the number of attributes which are sampled randomly as candidates at each split. Figure 4.8 shows the average AUC against the number of randomly selected predictors. The setting $mtry = 2$ gave the highest average AUC ($= 0.8429$).



(a) Closest Topleft Cut-off Point



(b) Youden Index Cut-off Point

Figure 4.9: ROC plot for Random Forest

Evaluation on Test Dataset:

For random forest, the AUC was 0.852(Figure 4.9). We found the closest topleft cut-off as 0.2370. At this cut-off, the sensitivity was 0.8537 and specificity 0.7986 (1− specificity = 0.2014), the distance was 0.2489.

The Youden index cut-off was 0.4780. At this cut-off, accuracy was 0.8486, with sensitivity = 0.4146 and specificity = 0.9722 (1− specificity = 0.0278)

Validation and Prediction on Mixed Dataset:

Random forest model predicted 157 cases to progress to ESRD = 1 within 10 years. It validated 59 cases as true positives, with sensitivity = 0.9365.

4.3.3 Boosting

We used the MBOOST [8, 29] package in R, which implements boosted generalized additive model.

Choosing the Model Parameters:

The **mstop** parameter specifies the number of boosting iterations. Figure 4.10 shows the average AUC against the number of randomly selected predictors. The setting $mstop = 150$ gave the highest average AUC (= 0.8575).

Evaluation on Test Dataset:

For Boosting, the AUC was 0.868(Figure 4.11). We noticed the closest topleft cut-off as 0.1734. At this cut-off, the sensitivity was 0.8293 and specificity 0.7778 (1− specificity = 0.2222), the distance was 0.2802.

The Youden index cut-off was 0.5696. At this cut-off, accuracy was 0.8541, with sensitivity = 0.4146 and specificity = 0.9792 (1− specificity = 0.0208)

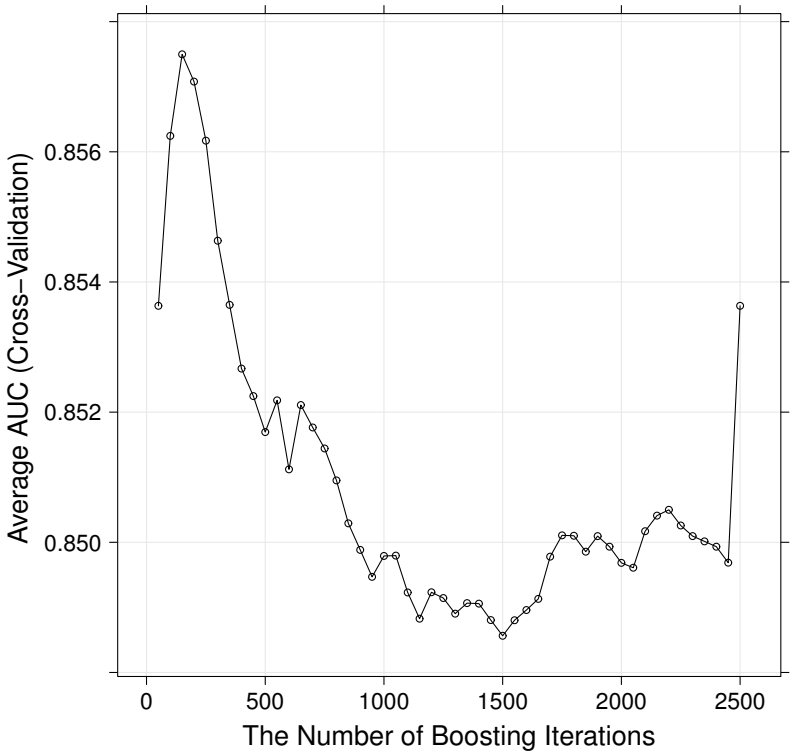


Figure 4.10: Average AUC Vs. Number of Boosting Iterations

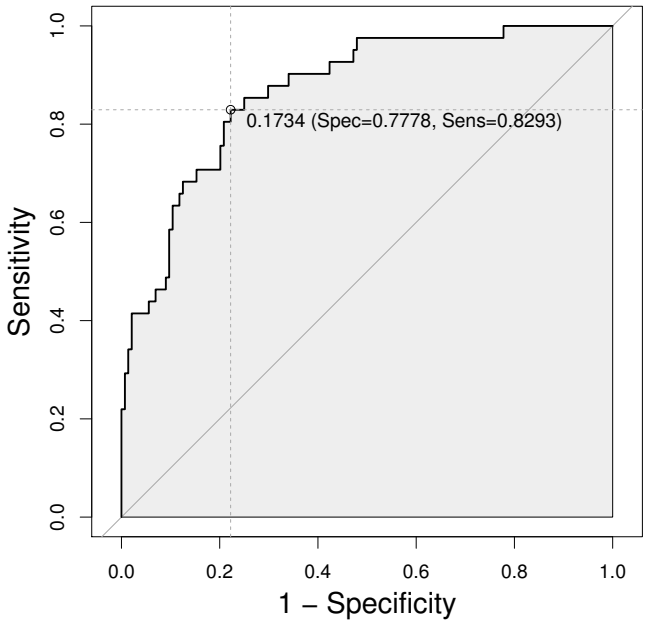
Validation and Prediction on Mixed Dataset:

The boosting model predicted 323 cases to progress to ESRD = 1 within 10 years. It validated 61 cases as true positives, with sensitivity = 0.9683.

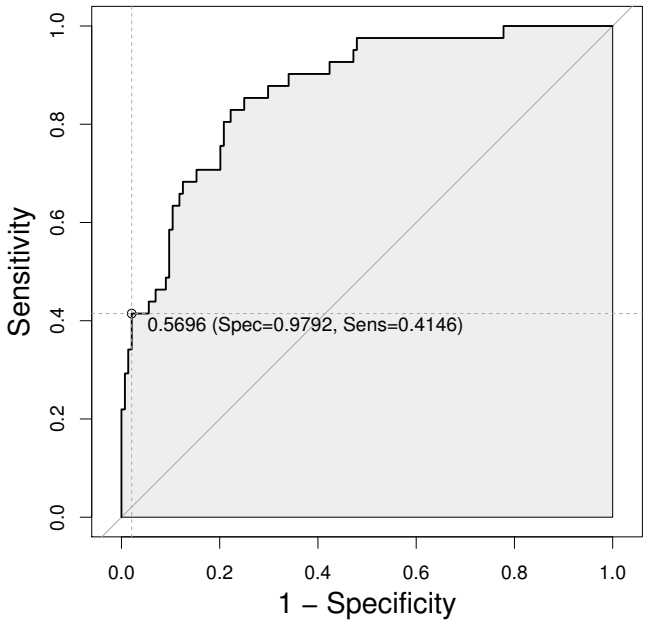
4.4 Model Assessment: Comparative Study

4.4.1 Test Dataset

We compared the performance (discriminatory ability) of the models generated from the learning algorithms on the test set. GFR (computed based on the attributes AGE and CR), SS% and GS% were the three common attributes chosen by all models. The measures such as distance from the top-left of ROC plot, maximum accuracy and AUC are shown in Table 4.8. All AUC estimates



(a) Closest Topleft Cut-off Point



(b) Youden Index Cut-off Point

Figure 4.11: ROC plot for Boosting

Table 4.8: Performance Comparison of Classifiers on Test Set

Classifier	AUC	Closest Top-left		Youden	
		Cut-off	Distance	Cut-off	Accuracy
Decision Tree	0.804	0.1291	0.3812	0.1810	0.800
Logistic Regression	0.840	0.1801	0.3379	0.6654, 0.7155	0.8486
Neural Network	0.834	0.1906	0.3275	0.6220	0.8486
Bagging	0.841	0.2200	0.3347	0.4600	0.8324
Random Forest	0.852	0.2370	**0.2489**	0.4780	0.8486
Boosting	**0.868**	0.1734	0.2802	0.5696	**0.8541**

Table 4.9: Performance Comparison of Classifiers on Mixed Dataset

Classifier	Sensitivity
Decision Tree	0.8730
Logistic Regression	**0.9841**
Neural Network	**0.9841**
Bagging	0.9365
Random Forest	0.9365
Boosting	0.9583

lay between 0.8 and 0.9, meaning that all models were good classifiers.

4.4.2 Mixed Dataset

We compared validation and prediction of the models on the mixed dataset. Table 4.9 shows that logistic regression model and neural network model validated the true positive cases with high sensitivity = 0.9841. The sensitivity measures of the other models were also good.

Among the 658 (ESRD = 0) cases in the mixed dataset, the decision tree predicted 311 cases, logistic regression 285 cases, neural network 217 cases, bagging 212 cases, random forest 157 cases, and the boosting 323 cases to progress to ESRD = 1 within 10 years.

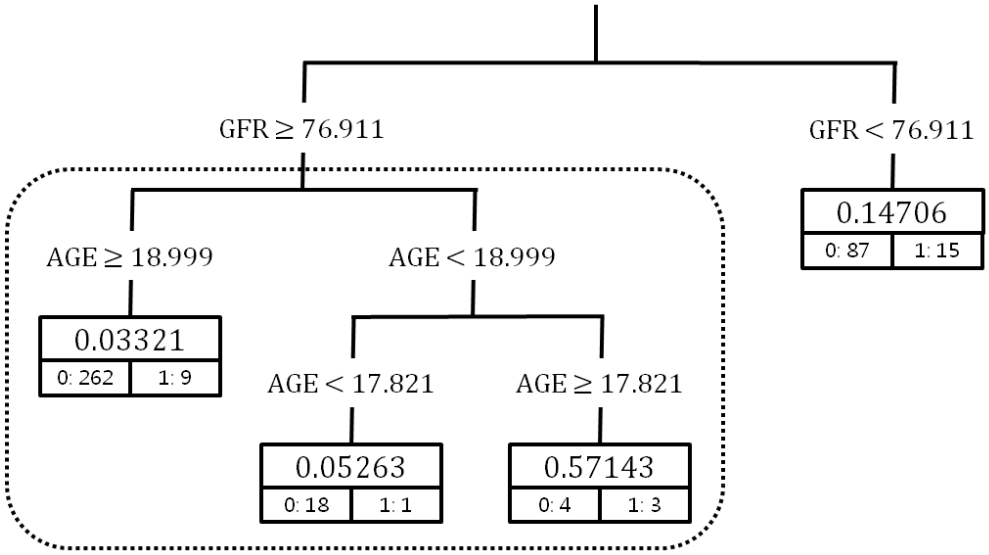
Chapter 5

Analysis

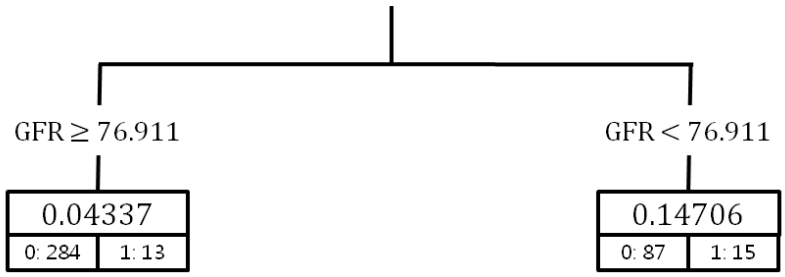
Decision trees are more comprehensible than other learning models. As they are commonly used by medical practitioners for diagnosis, we further analyse the results.

The decision tree of Figure 4.2 clearly indicates that the initial disease stage at first presentation is critical to the probability of progression to ESRD within 10 years. Low GFR and high 24-hour proteinuria (i.e. ineffective processing by the kidneys), high segmental glomerulosclerosis percentage (i.e. visible damage to cells on microscopic examination) and high percentage of global glomerulosclerosis (i.e. macroscopic damage) are all indicative of poor prognosis, as would be anticipated. The specific cutoff values can be useful to clinical practitioners.

When we observe the leftmost of tree in Figure 4.2, there are too specific conditions with AGE. We also notice that only few samples fell under the condition $AGE < 18.999$. This clearly indicates overfitting when cp is low, which controls the splits. To construct a more generalised and clinically useful model, we merged the rounded rectangular part with dotted line in Figure 5.1a and pruned the tree as in Figure 5.1b.



(a) Before Pruning



(b) After Pruning

Figure 5.1: Pruned Tree

Figure 5.2 shows the tree built with only the complete cases. We spotted an unexpected outcome in this model. It is the relationship with systolic blood pressure – almost inverse to the findings of other researchers [2]. Individuals presenting with low GFR but a relatively lower percentage of crescent cells and global glomerulosclerosis (i.e. worse processing by the kidneys, but less obvious damage) have better progression if they *are* hypertensive or prehypertensive. One tentative explanation is that these patients need higher renal perfusion to preserve their remaining kidney function, which higher blood pressure promotes.

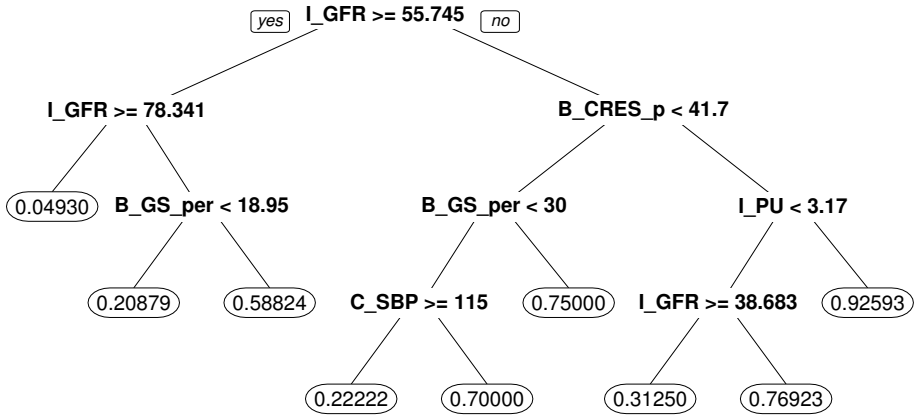


Figure 5.2: Classification Tree with Complete Cases

This unanticipated interaction between the variables certainly warrants further investigation. Potentially, it may lead to a reversal of the current blood pressure maintenance strategy for this group of patients, with substantial benefits for their lives.

Chapter 6

Summary and Conclusions

We built classifiers for predicting the probability of ESRD in IgAN patients within 10 years using individual learners such as decision trees, logistic regression and neural networks, and also ensemble learners such as bagging, random forest and boosting. All six classifier had good AUC performance, and provide useful information to the practitioner.

At the closest top-left cut-off value which maximize both sensitivity and specificity of each model, the classifier emphasised specificity in the case of bagging, and sensitivity in the other three cases. Thus, for both clinical application (e.g. in determining treatment) and consulting the patients, it would be important for decision making to be informed by all these results, depending on the relative weighting to be given to type 1 and type 2 errors.

In all six models, the presenting glomerular filtration rate, the extent of global glomerulosclerosis (macroscopic appearance) and the percentage of segmental glomerulosclerosis (microscopic appearance) are prognostically important. Patients who already have significantly impaired kidney performance generally have poorer outcomes; even when performance is not yet badly impaired, high levels of visible damage, either microscopically or macroscopically, indicate

poorer prognoses.

Of the learning methods we analysed, the logistic models showed the greatest sensitivity in validating the true positive cases in the mixed dataset, although the boosting gave the largest AUC. Differences between the classifiers were relatively small, but may be significant in individual cases.

Overall, the model based machine learning approach for predicting ESRD status of IgAN patients after a specific period can be useful for making medical and lifetime decisions.

6.1 Future Work

We applied single imputation methods to increase the data size and statistical power. However, there was small bias between imputed value and real value. To solve this problem, we will explore data more deeply and find relations among variables using domain knowledge and data transformation techniques. We will also explore multiple imputation which can include uncertainty of missing values to model.

We found expected and unexpected results by analysing the aforementioned machine learning models. The unexpected outcome may occur due to overfitting and it can be handled by pruning as in Section 5. If overfitting is not the cause, clinically significant facts can be discovered by intensive investigation. Hence, we will analyse more deeply and interpret models though they are complex.

The relatively small sample size emphasis it is not feasible, using these methods, to predict over substantially shorter or longer periods than 10 years. They also mean that, because patients initially present at very different stages of the disease, the training data is highly heterogeneous, leading to higher prediction errors. Finally, treating this problem as a classification problem from initial data means that subsequently accumulated data is not used. Thus, for a patient eight years out from initial diagnosis, all we can offer is the same prediction that was

given at the start – the highly informative subsequent progression of the GFR measurements cannot be used by the classifier.

One alternative approach, instead of modelling progression over a specific period using classification methods, is to probabilistically model the progression process itself. We are currently building a genetic programming system that learns probabilistic models describing the progression of the disease. If successful, this system should yield incremental probabilistic predictions, taking into account the progressive data for the patient, over a range of time periods.

Bibliography

[1] R. R. Andridge and R. J. Little. A review of hot deck imputation for survey non-response. *International Statistical Review*, 78(1):40–64, 2010.

[2] L. P. Bartosik, G. Lajoie, L. Sugar, and D. C. Cattran. Predicting progression in IgA nephropathy. *American Journal of Kidney Diseases*, 38(4):728 – 735, 2001.

[3] J. Berger and N. Hinglais. Intercapillary deposits of IgA-IgG. *Journal d'Urologie et de Nephrologie*, 74(9):694, 1968.

[4] J. Berger, H. Yaneva, B. Nabarra, and C. Barbanel. Recurrence of mesangial deposition of IgA after renal transplantation. *Kidney Int.*, 7(4):232–241, 1975.

[5] L. Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[6] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[7] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen. *Classification and Regression Trees*. CRC press, 1984.

[8] P. Bühlmann and T. Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, pages 477–505, 2007.

[9] S. Buuren and K. Groothuis-Oudshoorn. Mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45(3), 2011.

[10] I. J. Choi, H. J. Jeong, D. S. Han, J. S. Lee, K. H. Choi, S. W. Kang, S. K. Ha, H. Y. Lee, and P. K. Kim. An analysis of 4,514 cases of renal biopsy in Korea. *Yonsei Medical Journal*, 42(2):247–254, 2001.

[11] G. D'amico. The commonest glomerulonephritis in the world: IgA nephropathy. *Quarterly Journal of Medicine*, 64(3):709–727, 1987.

[12] A. J. Dobson. *An introduction to generalized linear models*. CRC press, 2001.

[13] G. D. Garson. Interpreting neural network connection weights. *Artificial Intelligence Expert*, 6(4):46–51, 1991.

[14] D. W. Joenssen. *HotDeckImputation: Hot Deck Imputation Methods for Missing Data.*, 2014. R package version 1.0.0.

[15] B. A. Julian, F. B. Waldo, A. Rifai, and J. Mestecky. IgA nephropathy, the most common glomerulonephritis worldwide: a neglected disease in the United States? *The American Journal of Medicine*, 84(1):129–132, 1988.

[16] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, volume 14, pages 1137–1145, 1995.

[17] A. Koyama, M. Igarashi, and M. Kobayashi. Natural history and risk factors for immunoglobulin A nephropathy in Japan. *American Journal of Kidney Diseases*, 29(4):526–532, 1997.

[18] Y. Lau, K. Woo, H. Choong, Y. Zhao, H. Tan, W. Cheung, and H. Yap. ACE gene polymorphism and disease progression of IgA nephropathy in Asians in Singapore. *Nephron*, 91(3):499–503, 2002.

[19] H. Lee, D. K. Kim, K.-H. Oh, K. W. Joo, Y. S. Kim, D.-W. Chae, S. Kim, and H. J. Chin. Mortality of IgA nephropathy patients: a single center experience over 30 years. *PloS one*, 7(12):e51225, 2012.

[20] A. Levey, J. Bosch, J. Lewis, T. Greene, N. Rogers, D. Roth, et al. Modification of diet in renal disease study group: A more accurate method to estimate glomerular filtration rate from serum creatinine: a new prediction equation. *Ann Intern Med*, 130(6):461–470, 1999.

[21] M. Levy and J. Berger. Worldwide perspective of IgA nephropathy. *American Journal of Kidney Diseases*, 12(5):340–347, 1988.

[22] L.-S. Li and Z.-H. Liu. Epidemiologic data of renal diseases from a single unit in China: analysis based on 13,519 renal biopsies. *Kidney International*, 66(3):920–923, 2004.

[23] A. Liaw and M. Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[24] J. Noh, D. Punithan, H. Lee, J. P. Lee, Y. S. Kim, D. K. Kim, and R. I. McKay. Predicting the progression of IgA nephropathy using machine learning methods. In *Proceedings of the 8th International Conference on Bio-inspired Information and Communications Technologies*, Dec. 2014.

[25] A. Peters and T. Hothorn. *ipred: Improved Predictors*, 2013. R package version 0.9-3.

[26] B. D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 1996.

[27] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine learning*, 37(3):297–336, 1999.

[28] T. Therneau, B. Atkinson, and B. Ripley. *rpart: Recursive Partitioning and Regression Trees*, 2014. R package version 4.1-8.

[29] W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S.* Springer, 2002.

요약

IgA 신염은 IgA 항체가 신장 사구체에 침착되면서 발생하는 염증이다. 이는 가장 흔한 사구체신염으로 우리나라를 비롯한 동아시아에서 특히 높은 유병률을 보인다. IgA 신염 환자는 평균 35세 전후로 젊고 말기신부전에 의해 개인적인 부담뿐만이 아니라 사회적, 경제적인 부담이 높기 때문에, IgA 신염 환자들을 위험도에 따라 분류하여 그에 따른 적절한 치료 방침을 세우는 것은 중차대한 과제라고 할 수 있다. 이미 IgA 신염의 결과를 예측하는 연구들이 기존에 있지만, 체계적이고 좋은 예측력을 갖는 방법은 부족한 상황이다. 우리는 본 연구에서 기계학습의 적용을 통해 새로운 예측 모형을 구축하는 것을 목표로 한다.

우리는 이를 위해 서울대학교 신경내과에서 1979년부터 2014까지 모은 자료를 기반으로 연구를 진행하였다. 자료에는 1622명의 환자들에 대한 90개 이상의 속성 정보가 들어있다. 우리는 이 중 17개의 속성들을 뽑아 예측 모형의 독립변수로 사용하였다. 하지만 이 속성들에 대해 하나 이상의 결측치를 가진 환자의 정보가 269개였는데, 이는 통계적 검정력의 큰 손실을 가져올 수 있다. 따라서 우리는 결측치 대체 방식을 이용하여 손실된 환자 정보를 복원하였다. 대체 방식의 결정을 위하여 평균값, 최빈값, 임의 대체와 같은 간단한 대체 방식을 기준으로 최근린 핫덱 대체와 연쇄식을 이용한 다변량 대체와 같은 더 복잡한 방식을 검증했다. 결과적으로 분류회귀나무를 이용한 다변량 대체가 가장 좋은 성능을 보였고 이를 적용하여 데이터를 최종 생성하였다.

위 데이터를 바탕으로 우리는 환자의 초기정보를 이용하여 10년 내에 말기신부전으로의 진행 여부를 예측하는 이진분류문제를 다뤘다. 이를 위해 다양한 기계학습법들이 적용되었는데, 의사결정나무, 로지스틱 회귀, 인공 신경망과 같은 단일 학습법을 비롯하여 배깅, 랜덤 포레스트, 부스팅의 앙상블 학습법을 사용하였다.

6가지 방식은 모두 시험 자료에 대해 0.804(의사결정나무)와 0.868(부스팅) 사이의 AUC 값을 가지며 좋은 성능을 보였다. 또한 해석력이 좋은 모형들을 분석

함으로써 예후 예측 인자들에 대해 예상했던 결과를 모형 내에서 볼 수 있었고, 더 나아가 인자들 간의 상대적 중요도나 인자 별 좋고 나쁨의 기준이 되는 값들을 확인할 수 있었다. 일부 환자들에 대해서는 예상치 못한 결과를 볼 수 있었는데 이러한 결과들에 대해 후속 연구를 진행함으로써 임상적으로 유의미한 사실을 발견할 수 있을 것으로 기대된다.

주요어: 면역 글로불린 A 신염 (IgAN), 말기 신부전 (ESRD), 결측치 대체, 기계 학습, 지도 학습, 앙상블 학습

학번: 2013-20786